

自主研究

開発言語の使用割合に着目した ソフトウェア生産性分析

開発言語の使用割合に着目したソフトウェア生産性分析

角田 雅照 奈良先端科学技術大学院大学／近畿大学
松本 健一 奈良先端科学技術大学院大学
大岩佐和子 一般財団法人 経済調査会 経済調査研究所
調査研究部 第二調査研究室 室長
押野 智樹 一般財団法人 経済調査会 経済調査研究所
調査研究部 第二調査研究室

はじめに

ソフトウェアを購入する企業にとって、ソフトウェアの価格は非常に重要である。企業が購入するソフトウェアとして、パッケージソフトウェアと受託開発ソフトウェアが存在する。パッケージソフトウェアの場合、市場に流通しているため、同等の機能を持つソフトウェア間の価格を比較し、価格の妥当性を判断することができる。これに対し、受託開発ソフトウェアの場合、受託者と委託者が個別に契約して作成されるため、他社で使われているソフトウェアの価格や機能を知ることができず、同等の機能を持つソフトウェアを比較して価格の妥当性を判断することが困難である。

受託ソフトウェア開発の価格妥当性の判断を支援するために、本稿ではソフトウェア開発工数や生産性に着目して分析する。開発工数とはソフトウェア開発の総作業時間であり、開発工数と技術者単価がソフトウェア価格の基礎となる。そのため、開発工数を増加させる（生産性を低下させる）要因は、ソフトウェア価格を上昇させる要因ともなる。

本稿では、開発工数（生産性）に影響する要因として、ソフトウェア開発で用いられるプログラミング言語に着目する。開発言語が開発工数に影響を与えることは、これまでの分析[1]で明らかにされている。ただし、以下の2つの点はこれまで明らかとなっていない。

- (1) 開発言語の使用割合と開発工数との関係
- (2) 開発工数の分析において、開発言語種別を用いた場合の効果

(1)について、開発言語は単独で用いることもあるが、複数を組み合わせて、例えば開発言語 Javaを90%、開発言語 PHPを10%用いることもある。このような開発言語の使用割合の変化により、開発工数がどのように変化するのはこれまで明らかでなかった。

(2)については、開発言語はJavaやC++はオブジェクト指向など、いくつかの種類に分類することができる。ただし開発言語にはDelphiなど、少数のプロジェクトでしか用いられていないものがあり、これらはプロジェクト数が少ないため、開発工数との関係を分析することが難しい。そこで本稿では、そのような開発言語を分析可能とするために、開発言語をいくつかの種別に分類した上で、開発工数との関係を示す。

1 分析方法

分析において、以下の統計的な手法を用いた。

有意確率：分析結果の確からしさを示し、一般に5%を下回る場合、結果が信頼できるといえる。

ヒストグラム：ある項目を、一定の値ごとに区切り、その区間に含まれるデータ件数を示した図表、項目の値がどのように分布しているかを視覚的に理解することができる。

重回帰分析：推定対象の項目A (**目的変数**) が、複数の項目B, C, D・・・ (**説明変数**) によりどの程度決定しているかを確かめるために用いる。言い換えると、項目B, C, D・・・により項目Aが推定可能かどうかを確かめるために用いる。

対数変換：ある項目に含まれる値それぞれについて対数を計算し、それを用いること。対数変換後に重回帰分析を行うことにより、極端に大きな値（外れ値）の影響を抑えることができる。

調整済R²：重回帰分析の結果から得られる。0から1の値を取り、1に近いほど、説明変数により目的変数が決定している、すなわち説明変数により目的変数が推定可能であることを示す。R²は一般に0.5以上が必要とされる。R²は説明変数の数が増えるほど

大きくなるため、説明変数の数を考慮して R^2 を調整したものが調整済 R^2 である。

標準化偏回帰係数：目的変数と、ある説明変数との関連の大きさを示し、値が大きいほど関連が強いことを示す。偏回帰係数の値が正の場合、説明変数の値が大きくなれば目的変数の値も大きくなることを示し、負の場合、説明変数の値が大きくなれば説明変数の値は小さくなることを示す。分析では目的変数を開発工数とするため、偏回帰係数の値が正ならば開発工数が増加する、すなわち生産性が低下することを示し、係数の値が負ならばその反対となる。

信頼区間：重回帰分析により求めた偏回帰係数は、現実のプロジェクトと完全に一致しているわけではない。ある説明変数の偏回帰係数は、実際には算出されたものよりも大きい可能性もあり、逆にもっと小さな可能性もある。これを考慮したものが偏回帰係数の信頼区間である。例えばある説明変数の偏回帰係数の信頼区間下限が0.02、上限が0.1となった場合、その変数の影響を最小に見積もるなら偏回帰係数を0.02、最大に見積もるなら偏回帰係数を0.1とする。

変数選択：重回帰分析の適用時に、目的変数と関連の弱い説明変数を除外する方法である。なお、重回帰分析では相互に関連の強い説明変数が含まれる場合、多重共線性が発生する。多重共線性が発生すると、偏回帰係数の正負が不正確になり、重回帰分析の結果を正しく評価できなくなる。多重共線性を避けるために、変数選択時には相互に関連の強い説明変数についても除外される。

中央値：値を大きい順に並べた場合に真ん中の順位となる値を示す。

相関係数：ある項目AとBとの関連の大きさを示し、値が大きいほど関連が強いことを示す。値が正の場合、項目Bの値が大きくなれば項目Aの値も大きくなることを示し、負の場合、項目Bの値が大きくなれば項目Aの値は小さくなることを示す。分析では外れ値に影響されにくい順位相関係数を用いた。順位相関係数は、各数値を大きさの順に順位で置き換えてから相関係数を算出する方法である。

2 分析に用いたデータ

分析対象としたデータは、経済調査会が2001から2018年度の「ソフトウェア開発に関する調査」で収集したプロジェクトデータ2,225件である。ここから、以下の条件を満たすプロジェクト601件を抽出した。

システム・ソフトウェア要件定義から総合テストまでの工程全てが実施されているもの（古いデータについては、基本設計以降の工程がすべて実施されているもの）

実績の開発工数、ソフトウェア規模、開発言語の使用割合が記入されているもの

上記のデータから、さらに生産性の上位2.5%と下位2.5%(合計5%。30件のプロジェクト)を外れ値とみなして除外した後、新規開発(再開発を含む)のプロジェクト450件を分析対象とした。

分析に用いた項目は以下のものである。

実績FP(ファンクションポイント)

実績工数

業種：以下のどの業種を対象にソフトウェアを開発したかを示す。

建設業、製造業、電気・ガス・熱供給・水道業、情報通信業、流通業、金融・保険業、サービス業、公務

システム構成：システムの構成は、以下のどれであったかを示す。

クライアントサーバシステム、Web系システム、メインフレームシステム、組み込み系システム

生産性変動要因：プロジェクトマネージャ(以下、PM)などが、対象プロジェクトに関して以下の項目を数値の1~5で5段階評価したもの。数値が小さいほど各項目の条件が厳しかったことを示す。

機能性、信頼性、開発スケジュール要求、発注要件の明確度と安定度、先行モデルの流用と標準モデルの採用、PMの経験と能力、アナリストの経験と能力、SE(システムエンジニア)・PG(プログラマ)の経験と能力

システムの社会的影響度：PMなどが、対象プロジェクトに関してシステムの社会的影響度を3段階評価したもの。数値が大きいほど影響度が大きいことを

図表 1 言語種別

言語種別	開発言語
アセンブラ	アセンブラ
手続型言語	FORTRAN, BASIC, COBOL, LISP, Pascal, C言語, PL/Iなど
オブジェクト指向言語	Java, C++, C# など
Web系言語1	ASP.NET, JSP, Perl, PHP, Python, Rubyなど
Web系言語2	マークアップ言語(HTML, XMLなど), JavaScript
問い合わせ言語	SQL (PL/SQLを含む), PostgreSQLなど
商用製品専用言語	VBA, ABAPなど
フレームワーク	.NET, Strutsなど
開発支援ツール	Biz/Browser, Silverlightなど
その他	

示す。

言語使用数: 対象プロジェクトにおいて使用されていた言語の数。例えばJavaとPHPを用いていた場合、言語使用数は2となる。

言語比率: 対象プロジェクトにおける、各開発言語の使用割合。ソフトウェアの規模に対して何割を各言語で開発したかを示す。例えばあるプロジェクトにおいて、ソフトウェアの80%をJava、20%をPHPで開発していた場合、Javaの言語比率は80%、PHPの言語比率は20%となる。

開発言語: ASP・ASP.NET, C, C# (VC#含む), C++ (VC++含む), COBOL, Delphi, HTML, Java, JavaScript, JSP, PHP, Ruby, SQL (PLSQL含む), VB, VB.NET, XML, MS-ACCESSなど。

種別比率: 開発言語をいくつかの種類で分類し、言語比率と同様の方法で使用割合を算出したもの。本稿では図表1に示すとおり、SPR社のプログラミング言語テーブル等を参考に10タイプの言語種別に分類した。例えばあるプロジェクトにおいて、ソフトウェアの80%を種別A、20%を種別Bで開発していた場合、種別Aの種別比率は80%、種別Bの種別比率は20%となる。

生産性: 開発工数をソフトウェアの規模(しばしばファンクションポイント法により測定される)で除した値。一般に、開発工数はソフトウェアの規模に伴って増加する。従ってソフトウェア開発の生産性分析は、規模以外で(生産性の分子である)開発工数、すなわち生産性を増減させる要因を中心に行われる。

変化率: 分析では、言語比率などの要因が変化すると、開発工数(生産性)がどの程度変化するかを表す

ことを目的に、変化率を定義した。例えばMS-ACCESSの言語比率が100%の場合と50%の場合の変化率は、モデルに基づいて(MS-ACCESSの言語比率に100%を代入して)算出した前者の開発工数を分母、後者の開発工数を分子として算出する。

3 生産性と言語の使用割合との関係

3.1 言語比率に基づく分析

言語比率が開発工数と関連しているか、すなわち使用している言語の割合によって生産性が変化するかを分析するために、重回帰分析を行った。重回帰分析の結果、ある言語の使用割合が高まるほど開発工数が増加する場合、その言語は生産性を低くしているといえる。逆に、使用割合が高まるほど開発工数が減少する場合、生産性を高くしているといえる。

重回帰分析の目的変数は開発工数である。説明変数として、実績FP、業種、システム構成、各プログラミング言語の言語比率、各生産性変動要因、言語使用数を用いた。これらはこれまでの分析[2]において、工数や生産性と関連が見られたものである。開発工数と実績FPは対数変換した。

説明変数の値が記録されているプロジェクト403件を重回帰分析した結果、調整済 R^2 は0.78となった。このことから適切なモデルが構築されているといえる。標準化偏回帰係数(以下、偏回帰係数)を図表2に示す。変数選択の結果、システム構成は説明変数に採用されなかったが、システム構成が開発工数に影響していないとは限らない。一般にシステム構成は信頼性

図表2 言語比率を用いた場合の標準化偏回帰係数

	実績FP	製造業	電気・ガス・ 熱供給・ 水道業	金融・ 保険業	サービス業	信頼性	機能性	PMの経験 と能力	言語使用数	MS-ACCESS 言語比率	C++ 言語比率
係数	0.78	-0.06	0.12	0.11	0.08	-0.11	-0.08	0.06	0.07	-0.05	0.05
p値	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.03	0.04

の要求や開発言語と関連が強いことが多いため、それらからシステム構成が一意に定まり、説明変数としては採用されなかった可能性もある。

図表2では実績FP(ソフトウェアの規模)の偏回帰係数が最も大きく、開発工数との関連が大きいことを示す。言語使用数の偏回帰係数が正の値であることから、言語使用数が多いほど開発工数が増加する。すなわち生産性が低下することを示している。これはこれまでの分析[1]と同様の傾向である。PMの経験と能力の偏回帰係数は正であり、これは経験と能力が高いほど生産性が低いことを示している。これについては、経験と能力が高いPMは難易度の高いプロジェクトを担当することが多いため、生産性が低下している可能性がある。

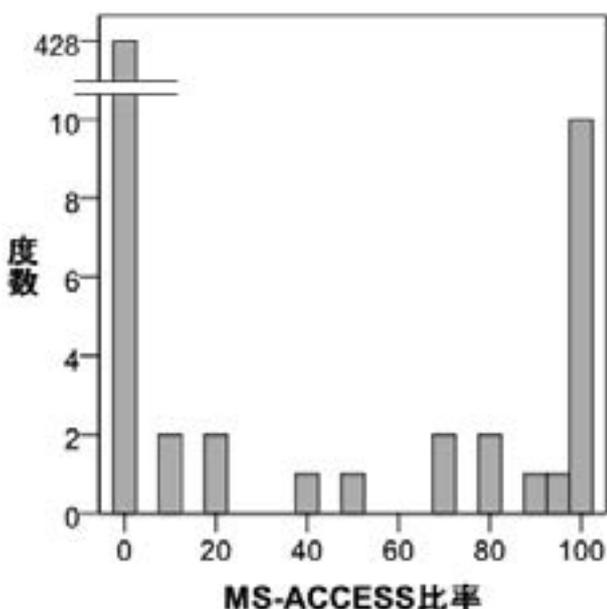
業種のうち、製造業の偏回帰係数が最も小さかったことから、製造業を対象としたソフトウェアは生産性が高くなりやすいことを示している。これも従来の分析[2]と同様の傾向である。信頼性と機能性については、偏回帰係数が負となっていた。これらの生産性変

動要因は数値が大きいほど条件が厳しくないことから、数値が大きい、すなわち条件が厳しくないが開発工数が小さくなり、生産性が高くなることを示している。

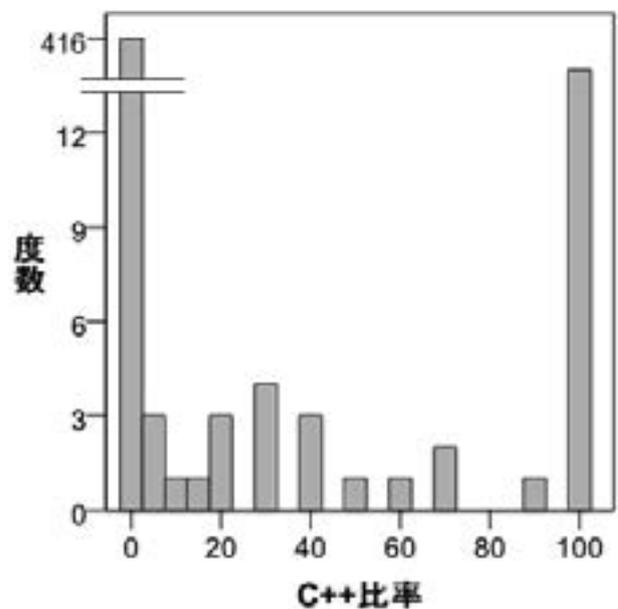
MS-ACCESSの言語比率の偏回帰係数は負の値、C++の言語比率の偏回帰係数は正の値となっていた。このことから、前者の言語比率が高い場合は生産性が高くなり、後者の言語比率が高い場合は生産性が低くなることを示している。例えばJavaの言語比率は含まれていないが、Javaの言語比率が100%だった場合、MS-ACCESSの言語比率は0%となり開発工数が小さくならないため、MS-ACCESSだけを使っていた場合よりも開発工数が増加する。より直感的に説明すると、MS-ACCESSは他の開発言語よりも生産性が高いといえる。逆に、C++は他の開発言語よりも生産性が低いといえる。

図表3、図表4にMS-ACCESSとC++の言語比率の分布を示す。図表3、4より、言語比率が0%以外のプロジェクト、すなわちそれぞれの言語を使っているプロ

図表3 MS-ACCESSの言語比率



図表4 C++の言語比率



プロジェクトでは、多くの場合は言語比率が100%となっていることがわかる。そこで、それらの言語を用いている場合と用いていない場合で、どの程度開発工数が異なるのかを、変化率(2章参照)を用いて確かめた。具体的には、作成されたモデルにおいて、両方の言語比率に0%を代入した場合(その他の開発言語を用いた場合)を基準とし、どちらか一方に100%を代入して開発工数を算出した。

変化率は、各言語比率を0%とした場合を分母、MS-ACCESS, C++それぞれの言語比率を100%にした場合を分子として求めた。変化率を図表5に示す。MS-ACCESSの変化率が0.67であることから、MS-ACCESSの言語比率が100%の場合、開発工数が0.67倍に減少すると推定される。逆にC++の言語比率が100%の場合、開発工数が1.44倍に増加する。

信頼区間を用いて、MS-ACCESSとC++の言語比率それぞれの開発工数への影響を、最小に見積もった場合(下限)と最大に見積もった場合(上限)を考慮した。図表5に示すように下限の場合、MS-ACCESS, C++とも変化率は1に近くなる。逆に上限の場合、前者の変化率は0.47、後者は2.04となる。すなわち、C++の言語比率が100%の場合、1.02倍しか開発工数が増加しない可能性があれば、2.04倍増加する可能性もある。このため、図表5の数値を絶対視すべきでないといえる。

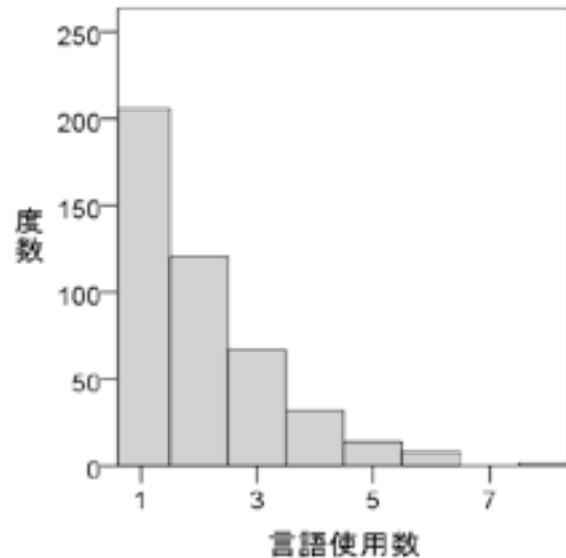
図表6に言語使用数の分布を示す。図表6より、分析対象プロジェクトの半数程度(約200プロジェクト)では、2つ以上の言語を用いているといえる。さらに約100プロジェクトが2つの言語を用いており、残りのプロジェクトでは3つ以上の言語を使用している。

そこで、言語使用数が1のプロジェクトを基準とし、使用数が2と3のプロジェクトではどの程度開発工数が異なるのかを確かめた。ここでは前者を変化率の分母、後者を分子とした。結果を図表7に示す。図表5で示した開発言語の変化率よりは、言語使用数の変化率は小さいといえる。

図表5 各言語比率が100%の場合の変化率と信頼区間

	下限	標準	上限
MS-ACCESS	0.95	0.67	0.47
C++	1.02	1.44	2.04

図表6 言語使用数の分布



図表7 開発言語数=1を基準とした場合の変化率

言語使用数 = 2	言語使用数 = 3
1.08	1.16

本節のまとめ：MS-ACCESSの比率が100%の場合、開発工数が33%減少する。C++の比率が100%の場合、開発工数が44%増加する。使用言語数が2になった場合、開発工数が8%増加する。

3.2 種別比率に基づく分析

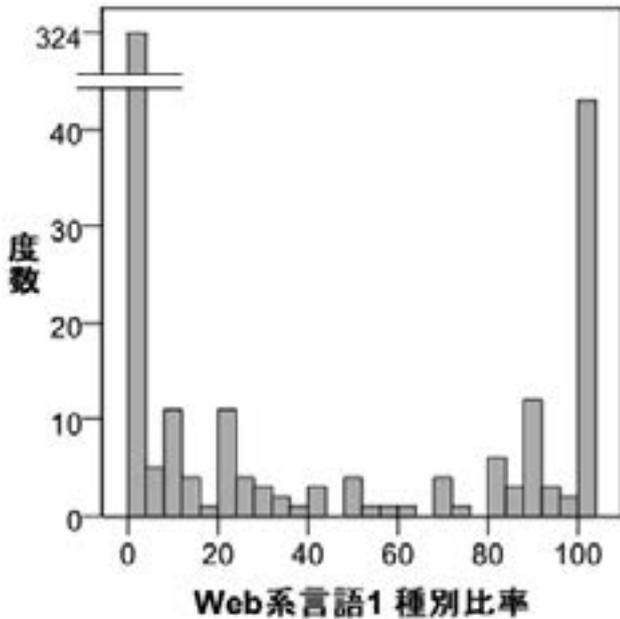
本節では、言語比率の代わりに種別比率を用いることができるかを確かめるために、言語比率の代わりに種別比率を説明変数に含め、開発工数を目的変数として重回帰分析を行った。重回帰分析の結果、ある種別の言語の使用割合が高まるほど開発工数が増加する場合、その言語種別は生産性を低くしているといえる。

種別比率以外の説明変数は、対数変換した実績FP、業種、システム構成、各生産性変動要因、言語使用数であり、これらは3.1節と同じである。説明変数の値が記録されているプロジェクト403件を重回帰分析した結果、調整済R²は3.1節と同じく0.78となった。こ

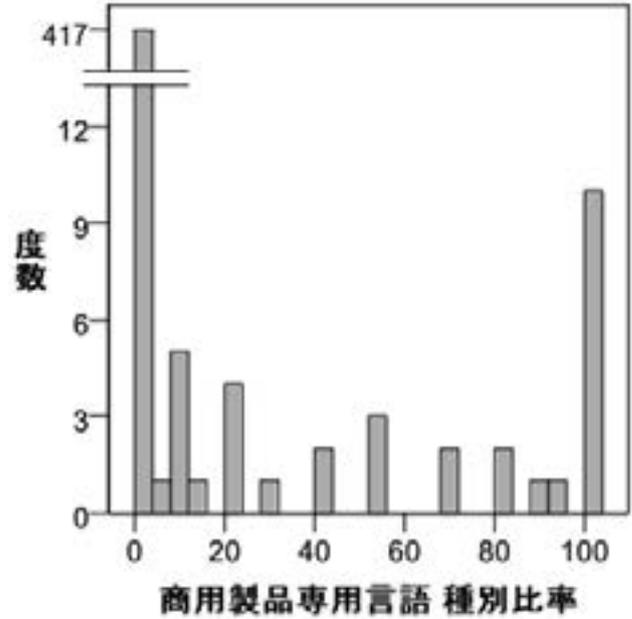
図表8 種別比率を用いた場合の標準化偏回帰係数

	実績FP	製造業	電気・ガス・ 熱供給・ 水道業	金融・ 保険業	サービス業	信頼性	機能性	PMの経験 と能力	言語使用数	Web系言語1 種別比率	商用製品 専用言語 種別比率
係数	0.77	-0.06	0.11	0.11	0.08	-0.11	-0.08	0.05	0.07	-0.05	-0.06
p値	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.03	0.02

図表9 Web系言語1の種別比率



図表10 商用製品専用言語の種別比率



のことから種別比率を用いても、モデルの説明力は変わらないといえる。偏回帰係数を図表8に示す。変数選択の結果、種別比率以外で採用された変数は3.1節と同じであり、また、偏回帰係数の値にも大きな差がなかった。

Web系言語1と商用製品専用言語の言語比率の偏回帰係数が負の値となっていたことから、これらの種別比率が高い場合は生産性が高くなるといえる。3.1節で述べたように、この結果は、これらの種別の言語が他の種別のものよりも生産性が高いことを示している。商用製品専用言語にはMS-ACCESSも含まれていることから、この種別比率については3.1節の結果と同様の傾向とみなせる。Web系言語1に含まれるものは、3.1節の分析において説明変数として採用されていなかったことから、こちらについては異なる結果であるといえる。

図表9、図表10にWeb系言語1と商用製品専用言語の種別比率を示す。図表9、10より、比率が0%となっているプロジェクト、すなわちそれぞれの言語を使っていないプロジェクトを除くと、比率が100%となっ

ていることが多いが、それ以外の比率のプロジェクトも一定数存在する。そこで、それぞれの種別比率が100%の場合と50%の場合で、どの程度開発工数が変化するかを確かめた。具体的には、作成されたモデルの種別比率に、100%と50%を代入して開発工数を算出した。なお、Web系言語1と商用製品専用言語を組み合わせ用いているプロジェクトは少なかったため、例えばWeb系言語1の種別比率に50%を代入する場合でも、商用製品専用言語の種別比率には0%を代入した。

各種別比率を100%とした場合を基準とした変化率を図表11に示す。図表11に示すように、それぞれの種別比率を50%にする（他の種別比率を高める）と、開発工数が1.1から1.2倍程度増加し、生産性が低下するといえる。

図表11 種別比率を100%から50%にした場合の変化率

Web系言語1	商用製品専用言語
1.10	1.23

本節のまとめ：Web系言語1の比率が100%から50%に低下した場合、開発工数が10%増加する。商用製品専用言語の比率が100%から50%に低下した場合、開発工数が23%増加する。

本章のまとめ：開発言語の代わりに言語種別に基づいたデータを用いても、見積り開発工数の誤差は大きくならない。

4 言語の使用割合に基づく工数見積り

3章で重回帰分析から得られたモデルの妥当性を、調整済R²以外の観点で評価するために、工数見積りを行った場合の誤差を評価した。見積り時の誤差が大きくなければ、モデルは適切に構築されているといえる。

見積り誤差は、リーブワンアウト法と呼ばれる方法により算出した。この方法では、例えば100個のプロジェクトのデータがある場合、99個のプロジェクトデータで重回帰分析を行い、残り1個のプロジェクトの工数を、重回帰分析で得られたモデルで見積る。そしてデータに記録されている実際の開発工数と、モデルから得られた開発工数との差を計算する。

誤差の評価基準として、以下の2つの平均値と中央値を用いた。

絶対誤差：実際の開発工数と、モデルから得られた開発工数との差

相対誤差：絶対誤差を、実際の開発工数とモデルから得られた開発工数のうち、小さい方の値で除したものを3.1節で構築したモデルと、3.2節で構築したモデルの誤差を**図表12**に示す。どちらのモデルも相対誤差の中央値が約50%であり、比較的誤差が小さいといえる。モデル間で誤差を比較すると。言語比率のほうの誤差が小さかったが、その差はわずかであった。このことから、3章で得られたモデルは妥当であり、かつ種別比率を言語比率の代わりに用いることができるといえる。

図表12 各モデルの見積り誤差

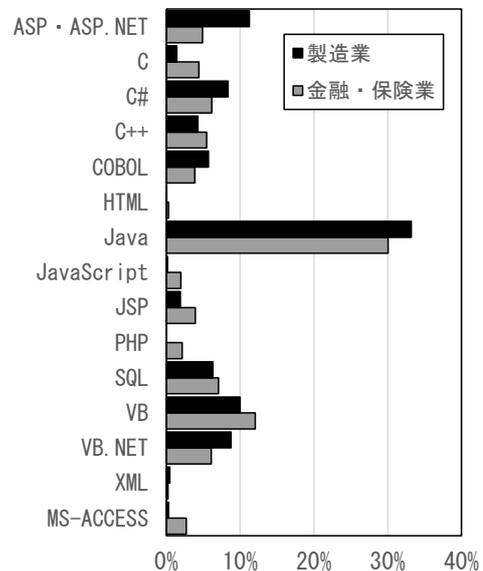
	言語比率モデル		種別比率モデル	
	絶対誤差	相対誤差	絶対誤差	相対誤差
平均値	50.0	82.6%	51.1	83.0%
中央値	14.5	53.9%	16.6	55.9%

5 業種で層別した場合の分析

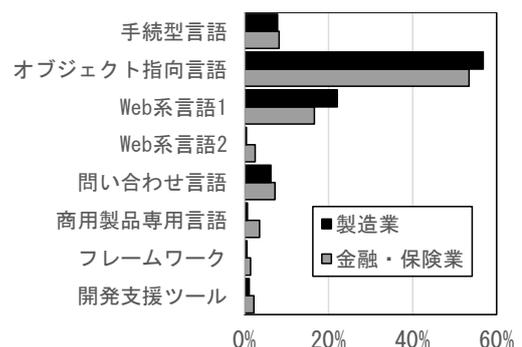
5.1 業種と開発言語の関連

業種と開発言語には関連が見られる（例えば金融・保険業ではCOBOLの使用率が高いなど）ことが多く、これまでの分析では業種によって生産性が異なっている[2]。そこで本章では、業種の影響を除外して言語比率と種別比率の生産性への影響を分析するために、データを業種で層別（特定の業種のデータのみを用いること）して分析した。3章の重回帰分析で採用された説明変数より、製造業、電気・ガス・熱供給・水道業、金融・保険業、サービス業が開発工数（生産性）との関連が強

図表13 業種別言語比率の平均値



図表14 業種別種別比率の平均値



いといえる。以降では誌面の都合上、プロジェクト数の比較的多かった製造業と金融・保険業でデータを層別する。プロジェクト数はそれぞれ95件と62件である。

図表13に業種別の言語比率の平均値を示す。例えば10件中8件のプロジェクトにおいてMS-ACCESSの言語比率が0%、残り2つのプロジェクトで100%ならば、平均値を20%としている。平均値の計算方法に注意する必要があるが、図表13から、例えばCは金融・保険業で利用される割合が高く、ASP・ASP.NETは製造業で利用される割合が高いことがわかる。開発言語ごとの比率の分布、例えば製造業においてASP・ASP.NETの言語比率が100%のプロジェクトがどの程度存在するかについては、以降の節で必要に応じてヒストグラムにより示す。

同様に、図表14に業種別の種別比率の平均値を示す。言語比率と比較すると、種別比率のほうが業種による違いが小さいといえる。

5.2 製造業における言語比率の影響

本節では、製造業のデータのみを用いて重回帰分析し、言語比率が開発工数に与える影響を分析した。重回帰分析で用いた説明変数は3.1節と同じものである。重回帰分析した結果、調整済 R^2 は0.79となった。

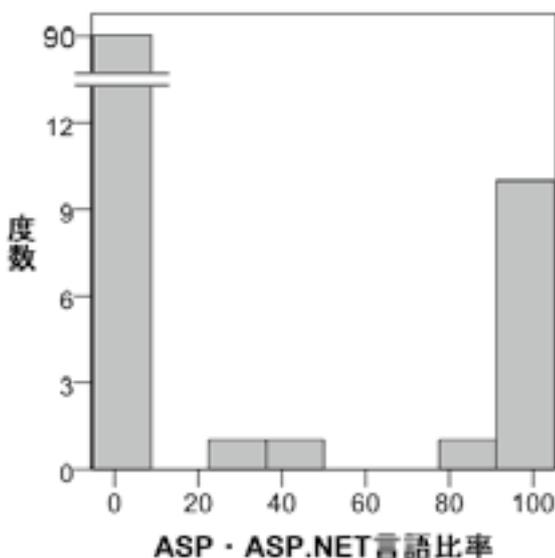
偏回帰係数を図表15に示す。変数選択の結果、開発言語に関する説明変数として、言語使用数といくつかの言語比率が採用された。ただし、Javascript, XML, MS-ACCESSについては、言語比率が0よりも大きいプロジェクトがごく少数であったため、製造業においてこれらの言語比率が生産性と関連があるかは確かではない。

言語使用数の偏回帰係数が正の値、ASP・ASP.NETの言語比率の偏回帰係数が負の値となっていたことから、言語使用数が多い場合は生産性が低くなり、ASP・ASP.NETの言語比率が高い場合は生産性が高くなるといえる。言語使用数については、3章の分析結果と同様の傾向である。なお、ASP・ASP.NETの言語比率が100%の場合に生産性が高くなる理由として、ASP・ASP.NETが対象とするシステムに対する要求が

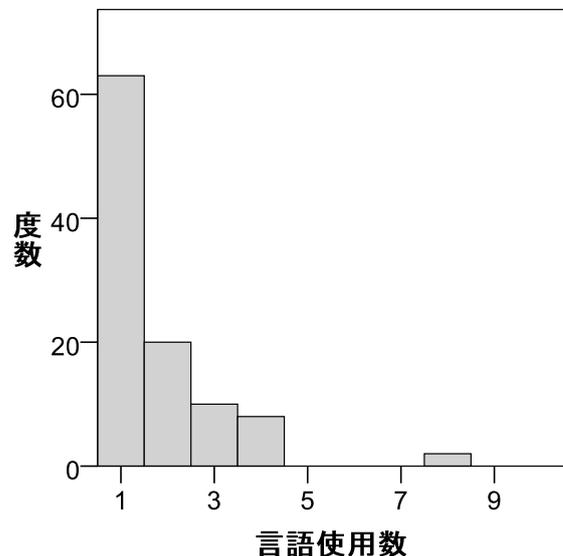
図表 15 製造業における言語比率を用いた場合の標準化偏回帰係数

	実績FP	先行モデルの流用と標準モデルの採用	開発言語使用数	ASP・ASP.NET言語比率	JavaScript言語比率	XML言語比率	MS-ACCESS言語比率
係数	0.83	-0.11	0.13	-0.11	0.17	-0.11	-0.07
p値	0.00	0.03	0.01	0.02	0.01	0.10	0.14

図表16 製造業におけるASP・ASP.NETの言語比率



図表17 製造業における言語使用数



影響していることも考えられる。システムの社会的影響度とASP・ASP.NETの言語比率の相関係数は弱い負の相関(-0.21)であり、言語比率が高いと社会的影響度が小さい傾向が少し見られる。

図表16にASP・ASP.NETの言語比率を、図表17に言語使用数を示す。図表16より、ASP・ASP.NETを使っているプロジェクトでは、ほとんどの場合で言語比率が100%となっていることがわかる。また図表17より、言語使用数が2以上のプロジェクトが一定数存在することがわかる。そこでASP・ASP.NETの言語比率が0%のプロジェクトと比較して、言語比率が100%の場合にはどの程度開発工数が変化するかを確かめた。また、言語使用数が1のプロジェクトと比較して、使用数が2のプロジェクトでは開発工数がどの程度増加するかを確かめた。

ASP・ASP.NETの言語比率が0%から100%に変化した場合の変化率と、言語使用数が1から2に変化した場合の変化率を図表18に示す。図表18に示すように、前者では開発工数が0.65倍に減少し、生産性が高まるといえ、後者では開発工数が1.14倍増加し、生産性が低下するといえる。3.1節の分析結果と比較すると後者の変化率がやや大きい、信頼区間を考慮すると大きな差ではないといえる。

図表18 製造業において言語比率などが異なる場合の変化率

ASP・ASP.NET言語比率 0→100	使用言語数 1→2
0.65	1.14

本節のまとめ：製造業においてASP・ASP.NETの比率が100%の場合、開発工数が35%減少する。使用言語数が1から2に変化した場合、開発工数が14%増加する。

5.3 金融・保険業における言語比率の影響

金融・保険業のデータのみを用いて重回帰分析し、言語比率が開発工数に与える影響を分析した。重回帰分析で用いた説明変数は前節と同じものである。重回帰分析した結果、調整済R²は0.89と比較的大きな値となった。

偏回帰係数を図表19に示す。アナリストの経験と能力の偏回帰係数は正であり、経験と能力が高いほど生産性が低いことを示している。この原因は、3.1節のPMの経験と能力と同様である可能性がある。変数選択により、言語使用数は採用されなかったが言語比率が複数採用された。前者については後者が前者の代わりとなり、採用されなかった可能性がある。なお、XMLについては、言語比率が0よりも大きいプロジェクトはわずかであったため、金融・保険業においてXMLの言語比率と生産性に関連があるかは不確かである。

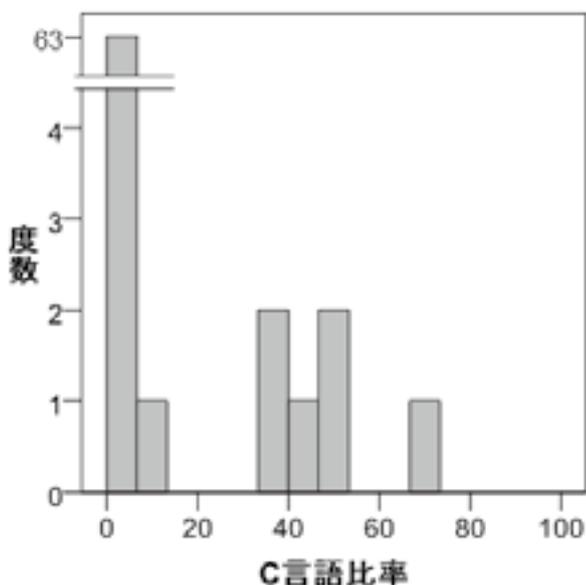
図表19より、C、JSP、VB、VB.NETは、各言語比率の偏回帰係数が負の値となっていることから、金融・保険業においては、これらの言語比率が高い場合は生産性が高くなるといえる。図表20から図表23に各言語比率の分布を示す。これらの図より、VB.NETを除き各言語比率は必ずしも高くなく、主開発言語として使われていないプロジェクトが多い。ただし後述する5.5節の分析では、金融業においても言語使用数が多いほど生産性が低い傾向が見られている。このことから、金融業では主開発言語以外にも開発言語を用いると生産性が低下するが、これらの言語を用いると生産性の低下が相対的に抑えられるとも考えられる。

上述の各言語比率の分布を考慮し、VB.NET以外については、各言語比率が0%の場合と30%の場合で、開発工数がどの程度異なるのかを確かめた。VB.NET

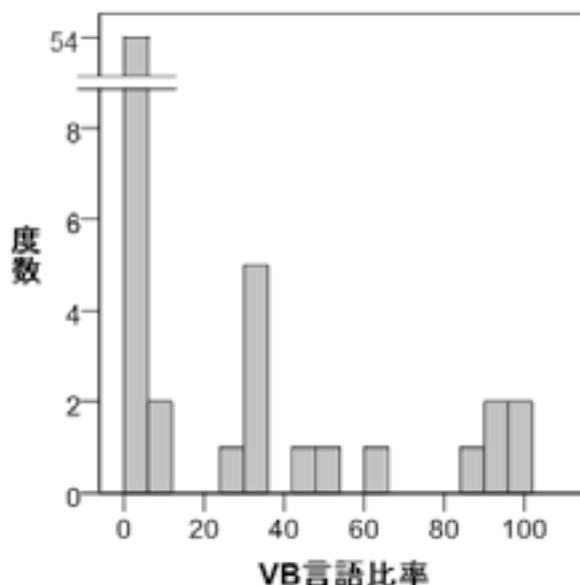
図表19 金融・保険業における言語比率を用いた場合の標準化偏回帰係数

	実績FP	信頼性	SE・PGの経験と能力	アナリストの経験と能力	C言語比率	JSP言語比率	VB言語比率	VB.NET言語比率	XML言語比率
係数	0.80	-0.22	-0.17	0.13	-0.09	-0.12	-0.12	-0.12	0.14
p値	0.00	0.00	0.00	0.01	0.05	0.01	0.01	0.01	0.00

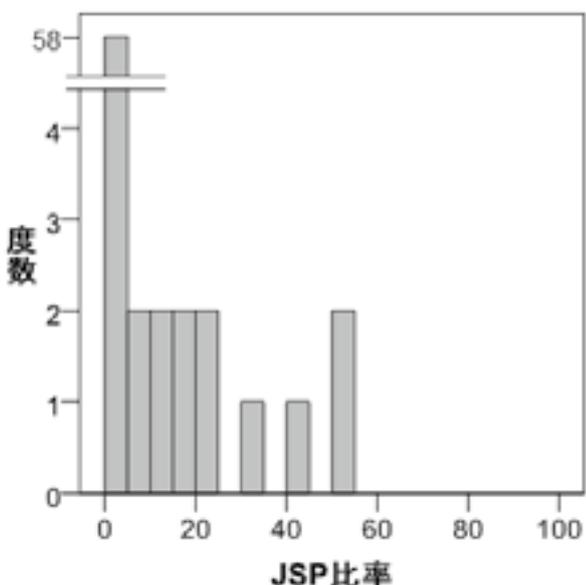
図表20 金融・保険業におけるCの言語比率



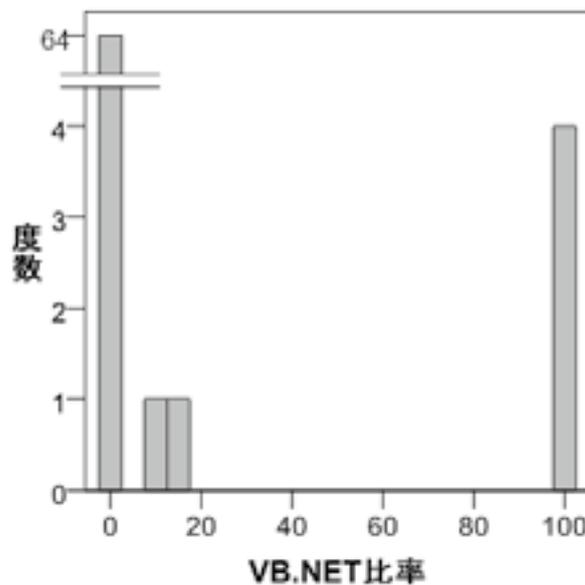
図表22 金融・保険業におけるVBの言語比率



図表21 金融・保険業におけるJSPの言語比率



図表23 金融・保険業におけるVB.NETの言語比率



については言語比率が0%と100%の場合で、開発工数の変化を確かめた。

それぞれの変化率を図表24に示す。VB.NET以外では、JSPの生産性が最も高かった。VB.NETの変化率は3.1節のMS-ACCESSの言語比率と比較しても、かなり大きいといえる。

本節のまとめ：金融・保険業においてC, JSP, VB, VB.NETを一定割合用いると、開発工数が16%から50%減少する。

図表24 金融・保険業において言語比率が異なる場合の変化率

C 言語比率 0→30	JSP 言語比率 0→30	VB 言語比率 0→30	VB.NET 言語比率 0→100
0.73	0.63	0.84	0.50

5.4 製造業における種別比率の影響

本節と次節では、種別比率に関して業種で層別して重回帰分析し、種別比率が開発工数に与える影響を分析した。本節では製造業のプロジェクトを対象とした。重回帰分析で用いた説明変数はこれまでの分析と同じものである。重回帰分析した結果、調整済R²は0.78となり、5.2節とほぼ同じとなった。

図表25 製造業における種別比率を用いた場合の標準化偏回帰係数

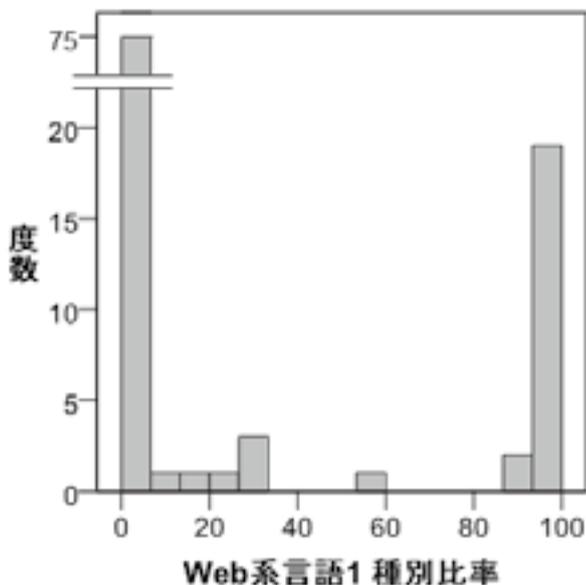
	実績FP	機能性	アナリストの経験と能力	先行モデルの流用と標準モデルの採用	開発言語使用数	Web系言語1種別比率	商用製品専用言語種別比率	開発支援ツール種別比率
係数	0.84	-0.08	0.07	-0.13	0.15	-0.07	-0.07	0.11
p値	0.00	0.13	0.20	0.01	0.00	0.19	0.16	0.04

偏回帰係数を図表25に示す。変数選択の結果、生産性変動要因については5.2節と一部異なるものが採用されたが、言語使用数は同様に採用された。種別比率は3個が説明変数として採用された。ただし、商用製品専用言語と開発支援ツールについては、種別比率が0よりも大きいプロジェクトがごく少数であったため、製造業においてこれらと生産性との関連は不明確である。Web系言語1にはASP.NETを含んでおり、かつ偏回帰係数も負の値であったため、5.1節の分析と同様の結果であるといえる。

図表26にWeb系言語1の種別比率を示す。図表26より、Web系言語1についても、多くのプロジェクトで種別比率が100%となっていることがわかる。そこでWeb系言語1の種別比率が0%と100%のプロジェクトで、どの程度開発工数が異なるのかを確かめた。また5.1節と同様に、言語使用数が1のプロジェクトと2のプロジェクトでは、開発工数がどの程度異なるのかを確かめた。

Web系言語1の種別比率が0%から100%に変化した場合の変化率と、言語使用数が1から2に変化した場合の変化率を図表27に示す。図表27に示すように、前

図表26 製造業におけるWeb系言語1の種別比率



者では開発工数が0.82倍に減少し、後者では開発工数が1.14倍増加するといえる。5.2節の分析結果と比較すると後者の変化率がほぼ同じであったが、前者は変化率が小さくなっていた。前者についてはASP・ASP.NET以外の開発言語の生産性が影響した可能性がある。

図表27 製造業において種別比率などが異なる場合の変化率

Web系言語1種別比率 0→100	使用言語数1→2
0.82	1.15

本節のまとめ：製造業においてWeb系言語1の比率が100%の場合、開発工数が18%減少する。使用言語数が1から2に変化した場合、開発工数が15%増加する。

5.5 金融・保険業における種別比率の影響

本節では金融・保険業のプロジェクトを対象として、種別比率の影響を分析した。重回帰分析で用いた説明変数はこれまでの分析と同じである。その結果、重回帰分析の調整済R²は0.86となり、5.3の結果よりわずかに低い値となった。

偏回帰係数を図表28に示す。変数選択の結果、生産性変動要因については5.3節と類似のものが採用された。また、5.3節の分析結果とは異なり、言語使用数が採用された。種別比率に関しては、商用製品専用言語のものが採用された。ただし、種別比率が0よりも大きいプロジェクトがごく少数であったため、金融・保険業における生産性との関連は不明である。

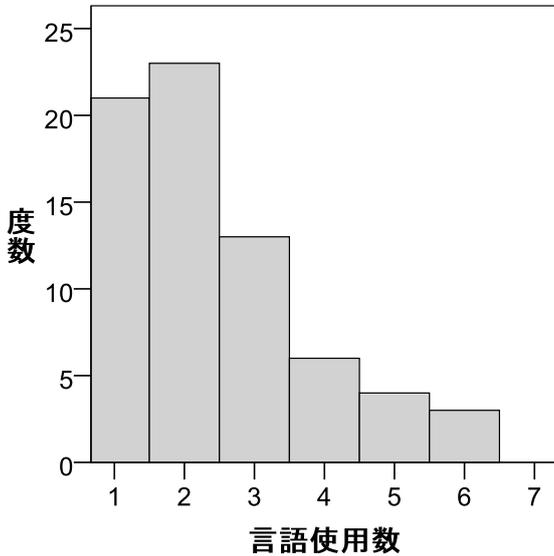
図表29に言語使用数の分布を示す。製造業の場合(図表17)と異なり、金融業では使用数が2のプロジェクトのほうが1のものよりも多かった。ここで言語使用数が1の場合を基準とし、使用数が2と3の場合でどの程度開発工数が異なるのかを確かめた。

結果を図表30に示す。図表30に示すように、言語

図表 28 金融業における種別比率を用いた場合の標準化偏回帰係数

	実績FP	信頼性	SE・PGの経験と能力	PMの経験と能力	言語使用数	商用製品専用言語種別比率
係数	0.85	-0.21	-0.17	0.12	0.07	0.09
p値	0.00	0.00	0.00	0.04	0.16	0.07

図表 29 金融・保険業における言語使用数



図表 30 金融・保険業において言語使用数異なる場合の変化率

使用言語数 = 2	使用言語数 = 3
1.08	1.16

使用数の増加に従い、変化率も増加している。3.1節での同様の分析(図表7)では業種を限定していなかったが、変化率の値については、この結果とほぼ同じであった。

本節のまとめ：金融・保険業において使用言語数が増加するごとに、開発工数が約8%以上増加する。

おわりに

本稿では、開発言語の使用割合に着目し、ソフトウェアの開発工数(生産性)に影響する要因を分析した。少数のプロジェクトでしか用いられていない開発言語についても分析するため、開発言語をいくつかの種別に分類した場合についても分析した。分析の結果、以下の傾向が見られた。

- MS-ACCESSの比率が100%の場合、開発工数が33%

減少する。C++の比率が100%の場合、開発工数が44%増加する。使用言語数が2になった場合、開発工数が8%増加する。

- Web系言語1の比率が100%から50%に低下した場合、開発工数が10%増加する。商用製品専用言語の比率が100%から50%に低下した場合、開発工数が23%増加する。
- 開発言語の代わりに言語種別に基づいたデータを用いても、見積り開発工数の誤差は大きくならない。また、業種ごとにデータを分析した結果、以下の傾向が見られた。

- 製造業においてASP・ASP.NETの比率が100%の場合、開発工数が35%減少する。使用言語数が1から2に変化した場合、開発工数が14%増加する。
- 金融・保険業においてC、JSP、VB、VB.NETを一定割合用いると、開発工数が16%から50%減少する。
- 製造業においてWeb系言語1の比率が100%の場合開発工数が18%減少する。使用言語数が1から2に変化した場合、開発工数が15%増加する。
- 金融・保険業において使用言語数が増加するごとに、開発コスト開発工数が約8%以上増加する。

なお、3.1節で述べたように、分析結果の解釈については信頼区間を考慮すべきである。例えばMS-ACCESSの比率が100%の場合、開発工数が5%しか減少しない可能性もあれば、53%減少する可能性もある。よって、上記で記載されている数値は絶対視すべきでなく、あくまで参考値とすべきであるといえる。

参考文献

- [1] 大岩佐和子 押野智樹 “開発言語が生産性に与える影響の分析” 経済調査研究レビュー Vol.18 pp.92-103 (2016)
- [2] 大岩佐和子 押野智樹 中井聡美 “ソフトウェア開発データリポジトリの分析” 経済調査会経済調査研究所 (2020)